

Early Chronic illness Detection

Satyam (2021284)

Aditya Jain (2021305)

Suyash Kumar (2021293)

Vickey Kumar (2021299)



INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI



Addressing Rising Chronic illness Rates

- Globally, An estimated 3.8% of the population experience depression, including 5% of adults, and 5.7% of adults older than 60 years. [1]
- India's **suicide rate** is among the highest globally, with an estimated **21.1 suicides per 100,000 people**, indicating the severe impact of untreated mental health issue.[2]

The Impact of Mental Health Issues

- Depression is a leading cause of disability worldwide, contributing significantly to the global burden of disease.
- The **Indian Council of Medical Research (ICMR)** highlighted that mental health disorders, including depression, are pushing around **20% of Indian households** into poverty due to high healthcare costs.

Importance of Timely Detection

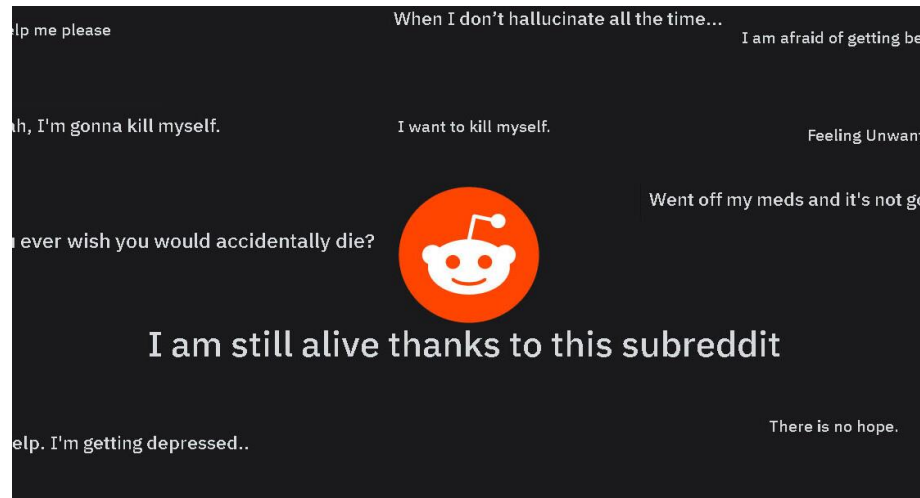
- **Prevention of Severe Outcomes:** Timely detection of depression can reduce the risk of severe mental health conditions, such as chronic depression or suicide
- **Cost-Effective Treatment:** Early identification allows for less intensive and more cost-effective interventions, reducing healthcare expenses and improving long-term outcomes for individuals [3]

Motivation



How we use in Real Life

- We have created a model that collects individual data through a Google form by collaboration with college wellbeing cells. Google forms covering questions like lifestyle and family history, to predict chronic health conditions.
- Based on the predictions, we provide personalized health recommendations and share doctor contact information for early intervention.



Depression Detection Using ML (Chauhan et al. 2023):

- Establishes a relationship between stress-related factors and depression detection using machine learning techniques.
- Stress related factors are qualitatively.
- This study focuses on identifying individuals who may not outwardly show signs of depression but are under stress.

Neuroimaging-Based Detection (Fu et al. 2018):

- ML analysis of neuroimaging data identified structural and functional brain changes related to depression, with an accuracy of 86%.
- This method use CV to analyze neuroimaging based detection

Speech and Language Patterns for Depression Detection (Alghowinem et al. 2016):

- ML algorithms analyze speech features like slower speech rate, longer pauses, and lower pitch, achieving an accuracy of 80% in predicting depression.
- The SVM model and NLP was effective in identifying distinct speech patterns linked to depressive symptoms

Concluding Remarks:

- Our project focus on - Integration of Additional Factors: Future research could integrate more comprehensive datasets, including historical markers (e.g., Family History of Depression, Mental health and substance) and daily activity factors(sleep pattern), to enhance the prediction accuracy.

Dataset Overview

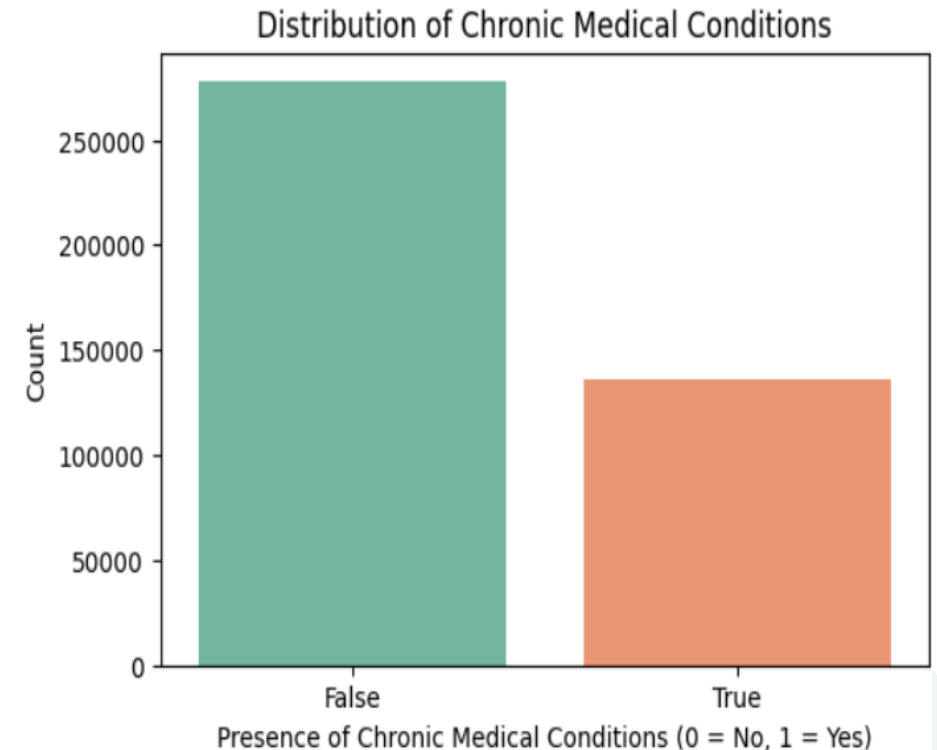


Data Collection:

- Data sourced from Kaggle {**Author - Anthony Therrien**}, i.e [Dataset](#)
- This dataset contains information on individuals with various attributes related to their personal and lifestyle factors. It is designed to facilitate analysis in areas such as health, lifestyle, and socio-economic status..

Demographic of Dataset Entries:

- Data consists of **413768** entries with label 'No' or 'Yes' for Chronic Disease and **16 columns**



Dataset Overview



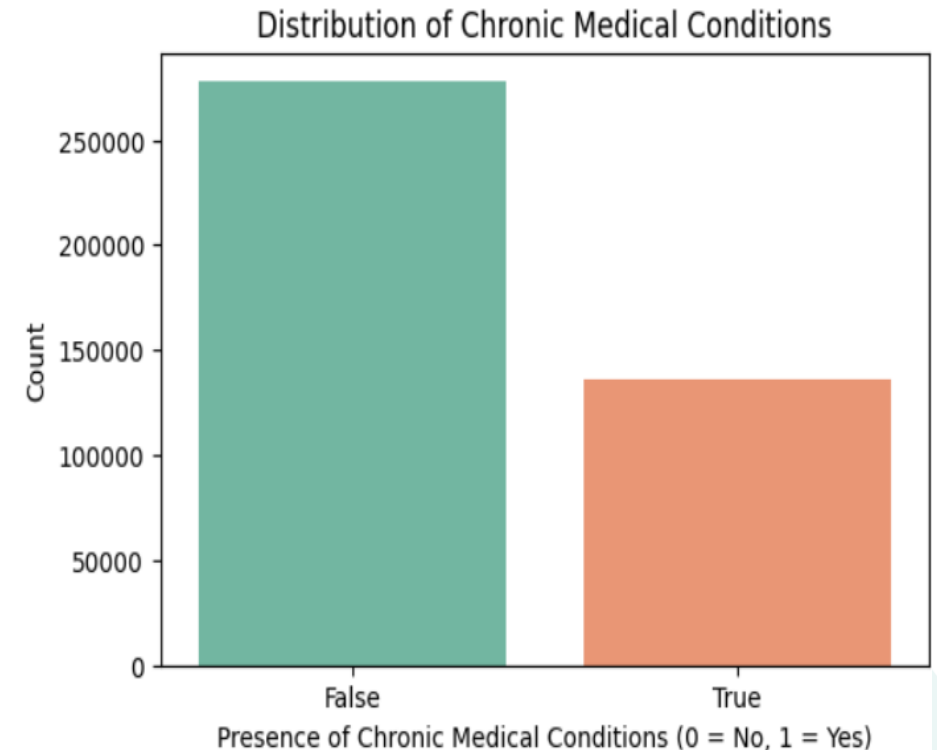
Data Attributes

The dataset contains the following attributes

- **Demographic** factors (e.g., age, marital status, education)
- **Health and lifestyle** indicators (e.g., smoking status, physical activity level, sleep patterns)
- **Medical and Psychological history** (e.g., mental illness, substance abuse, family history of depression)
- **Socio-economic** variables (e.g., income, employment status).

Demographic of Dataset Entries:

- Data consists of **413768** entries with label **‘No’** or **‘Yes’** for Chronic Disease and **16 columns**



Data Preprocessing & EDA



Data Pre-Processing:

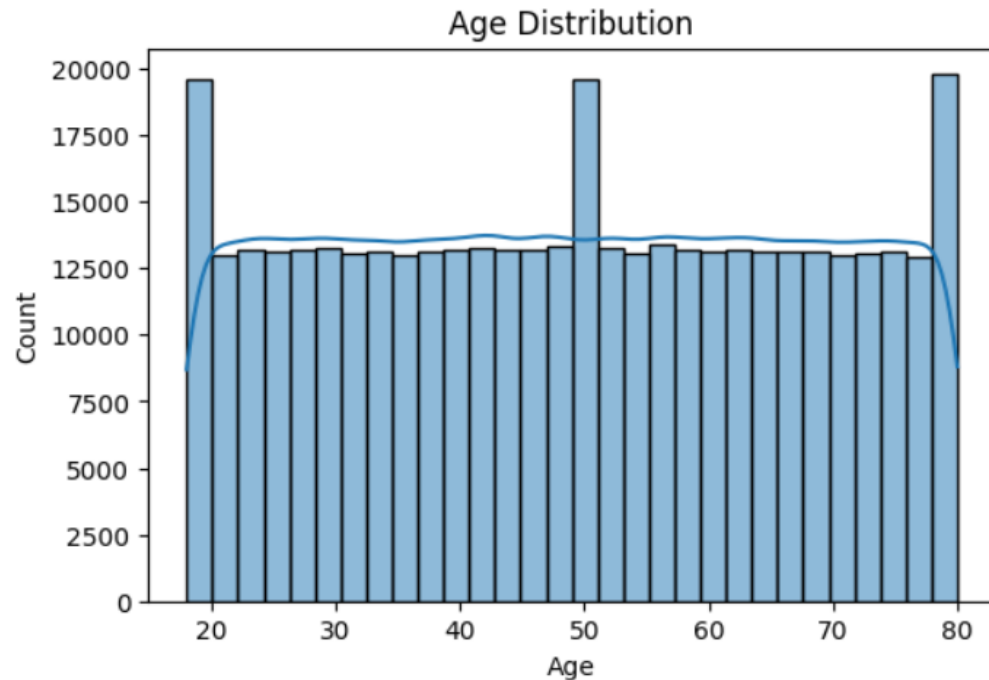
- **Class imbalance** was addressed using SMOTE to balance "No" and "Yes"
- **Standardization** minor tweaking is expected to be scaled
- **One-hot encoding** for categorical variables like Marital Status, Education Level, and Smoking status etc
- **Missing Value**
- **Non- Linear Relationships**



Exploratory Data Analysis:



Data Analysis

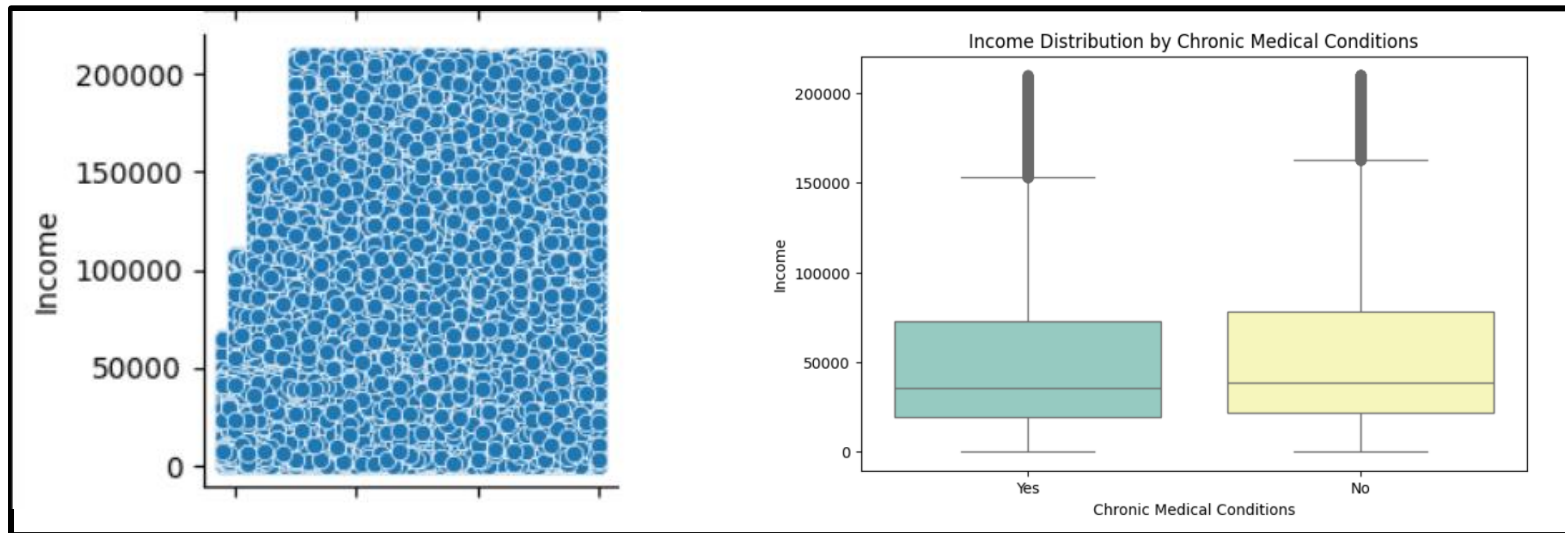


- Age distribution shows peaks at 20, 50, and 80, suggesting targeted survey samples.
- Relationships between age, income, lifestyle, and chronic conditions are likely non-linear in health data..

Exploratory Data Analysis:



Income vs Age & Income vs Chronic medical condition Analysis



- **Income distribution:** Income seems to have a broad spread across all ages, ranging from 0 to over 200,000. There is no strong visual correlation between age and income.
- **Income clustering:** The income distribution is relatively dense across higher age groups, with no significant clustering in lower age groups.

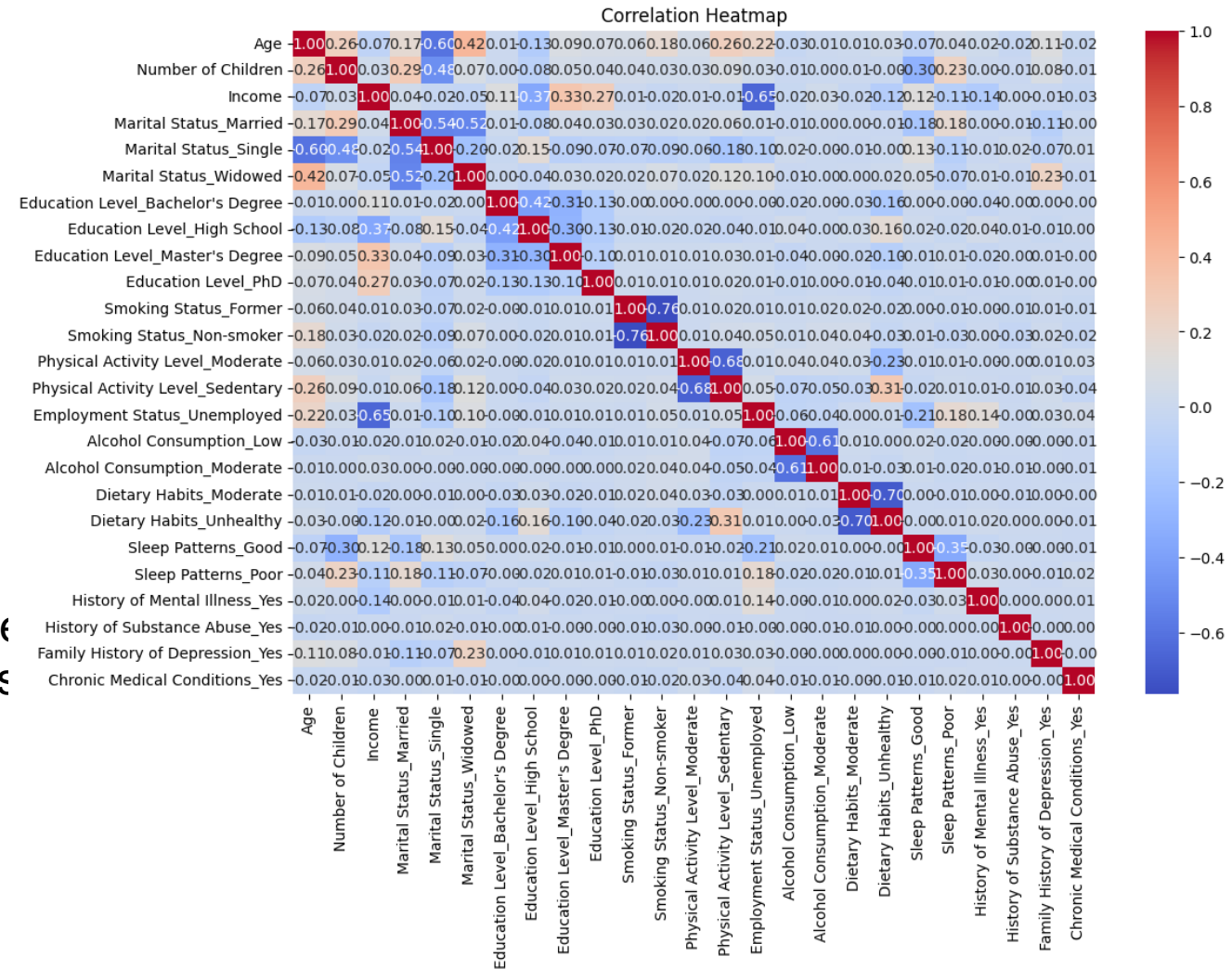
Exploratory Data Analysis:



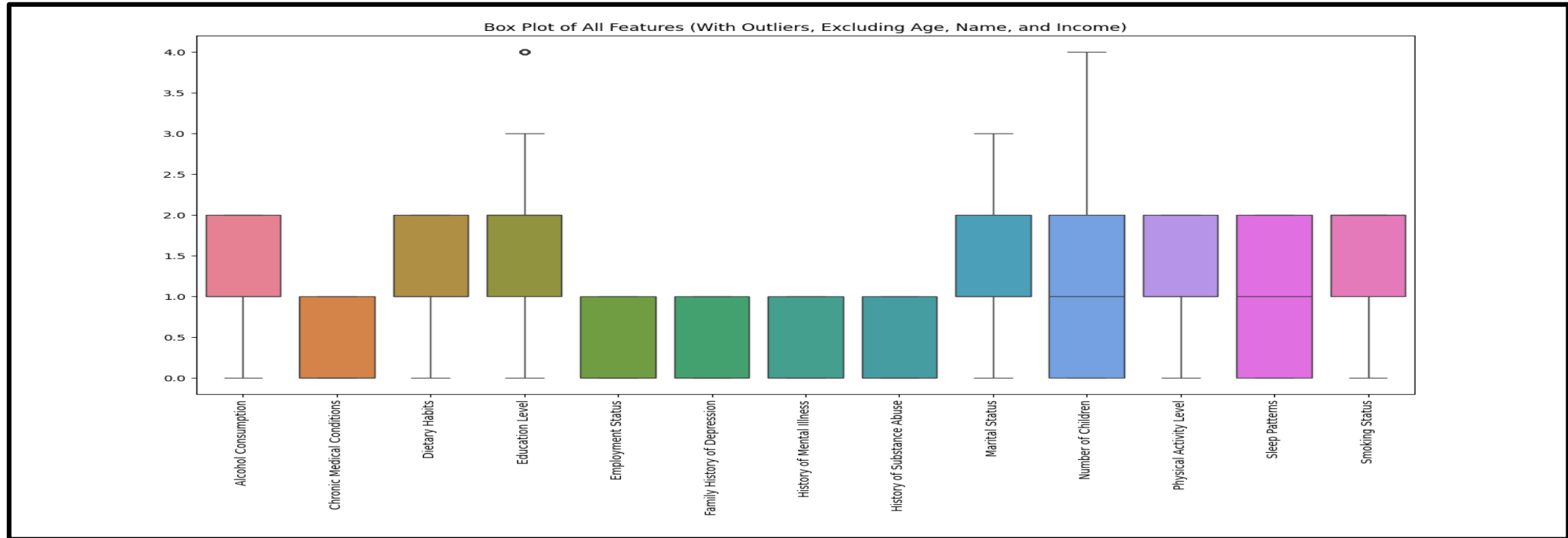
Heat Maps

The heatmap shows correlations between various factors in the dataset.

- **Positive correlations** (in red) indicate that as one factor increases, so does the other, such as income and employment or physical activity and good sleep.
- **Negative correlations** (in blue) show an inverse relationship, where an increase in one factor leads to a decrease in another, such as unemployment with income or poor sleep patterns with physical activity and healthy habits.



Outlier Detection



Boxplot to detect the outliers in the data. we can clearly see that there are many sample points in the dataset that show the central tendency of the data hence they do not display prominent outliers, meaning their data distributions are more contained within the range represented by their whiskers.

Methodology – Logistic Regression

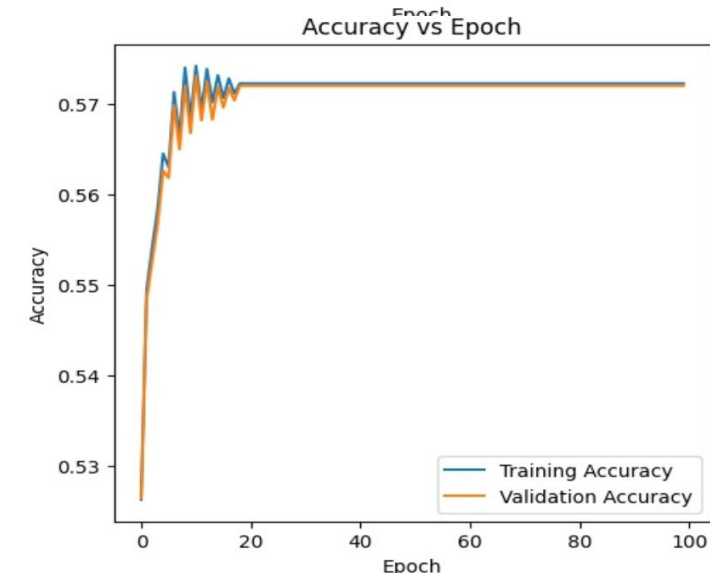
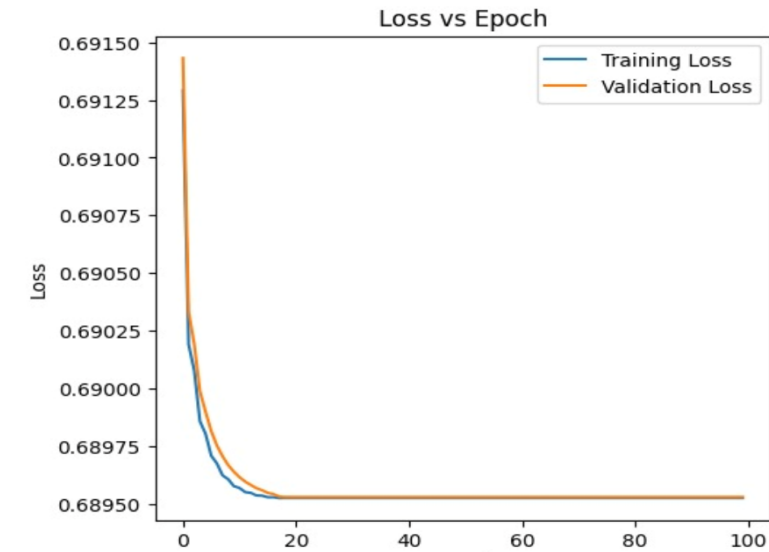


Logistic Regression with Batch Gradient Descent (BGD) – 57%

- BGD ensures more stable updates by computing gradients over the entire dataset, leading to better convergence.
- Handles class imbalance effectively with balanced class weights, contributing to moderate performance.

Logistic Regression with Stochastic Gradient Descent (SGD) – 52%

- SGD updates model parameters with each sample, resulting in noisier updates and potential instability.
- Performance is lower due to sensitivity to learning rate and less precise convergence compared to BGD.



Methodology – Why Batch gradient > SGD ?

Stability of Updates:

- **SGD** is more **sensitive to learning rate** because it updates the weights frequently (one sample at a time). If the learning rate is not tuned correctly, it can result in **poor convergence or oscillations**.
- In **BGD**, the weights are updated after processing the entire dataset, making it **less sensitive to learning rate adjustments**.

BGD achieves **better convergence** since it minimizes the loss more consistently, especially on smaller or well-conditioned datasets.

Sensitivity to Learning Rate:

- **Batch Gradient Descent Logistic Regression** integrates **L2 regularization** by default, which helps prevent overfitting.
- **SGDClassifier** requires additional parameter tuning to achieve optimal regularization. If not properly tuned, it may **overfit or underfit** the data.

BGD can achieve **better results** even with default hyperparameters, while **SGD may require careful tuning** of learning rate or other hyperparameters.

Methodology – Performing RFE to find Best Top Features & L2 Regularization



L2 (Ridge) Logistic Regression with Recursive Feature Elimination (RFE) On top 5 features– 56%

L2 (Ridge) Logistic Regression with Recursive Feature Elimination (RFE) On top 10 – 57%

Selected Feature of RFE : Education Level, Smoking Status, Physical Activity Level, Employment Status, Alcohol Consumption, Dietary Habits ...etc



Methodology – Why RFE + L2 Regularization has better Accuracy??




- RFE selects the most informative features, enhancing performance by reducing dimensionality.
- Ridge regularization prevents overfitting, leading to better generalization.



Purpose of PCA (Principal Component Analysis)

- PCA is a **dimensionality reduction technique** that transforms the original features into a smaller set of components, retaining as much variance (information) as possible. This is useful to simplify models and reduce overfitting.

Logistic Regression with PCA – 54%

- PCA simplifies the data by retaining only the most significant components, reducing noise.
 - However, some relevant feature information might be lost during dimensionality reduction, slightly impacting performance.
- 

Methodology – Logistic Regression with SMOTE and Optimal Threshold :



Purpose of SMOTE

- SMOTE **balanced** the dataset, which ensures that the model will give **equal importance** to both classes, improving performance for the minority class and preventing biased predictions.

Purpose of Threshold

- By **tuning the decision threshold**, we move away from the default 0.5 threshold to one that optimizes the model's performance based on the ROC curve.

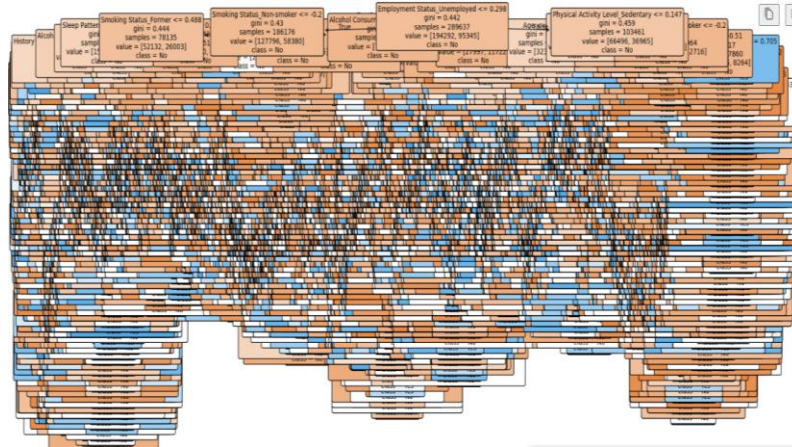
Logistic Regression with SMOTE and Optimal Threshold – 60%



Methodology – Decision Tree



Without
PRUNING



With Pruning

1. Decision Tree (Gini Criterion) – Accuracy 57%

1. Captures non-linear relationships but lacks robustness in handling complex interactions compared to other ensemble methods.

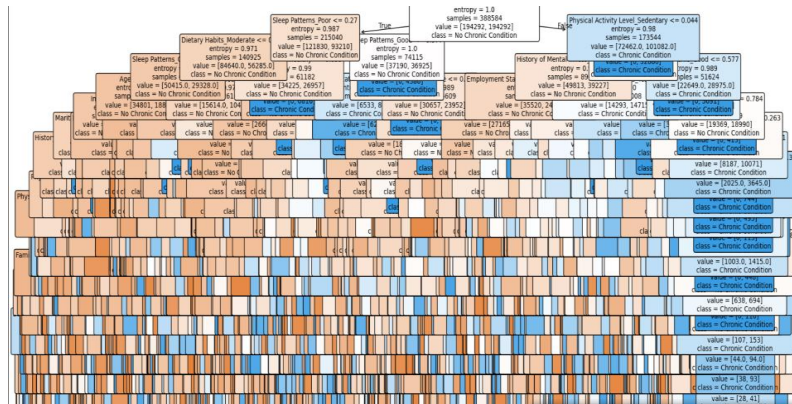
2. Decision Tree (Entropy Criterion) – Accuracy 63%

1. Entropy-based splitting captures finer information gain, improving the model's ability to separate classes better than Gini.

Without Pruning Accuracy : 55%

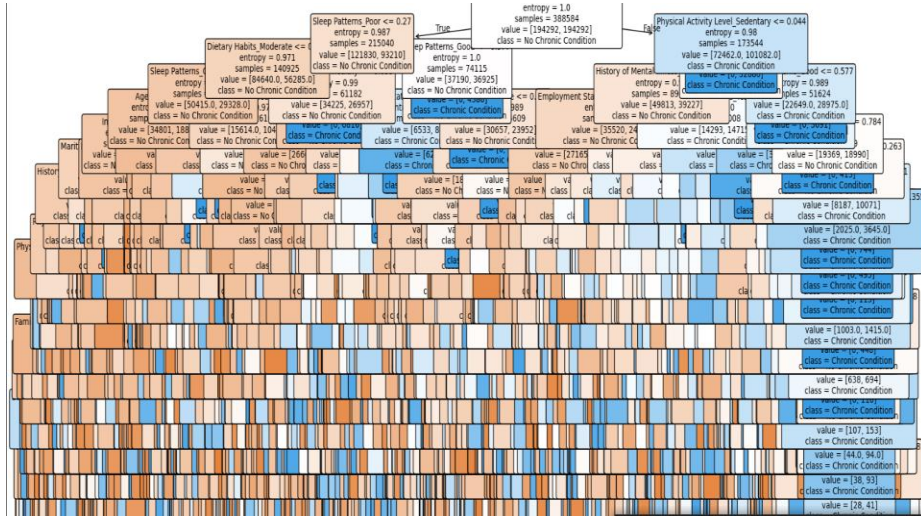
Entropy > Gini

With
PRUNING



t-SNE

Methodology – Decision Tree With Pruning



Perform Hyperparameter Tuning Using **GridSearchCV** + **SMOTE**-Balanced Training Dataset to find the Best Combination of Hyperparameters.

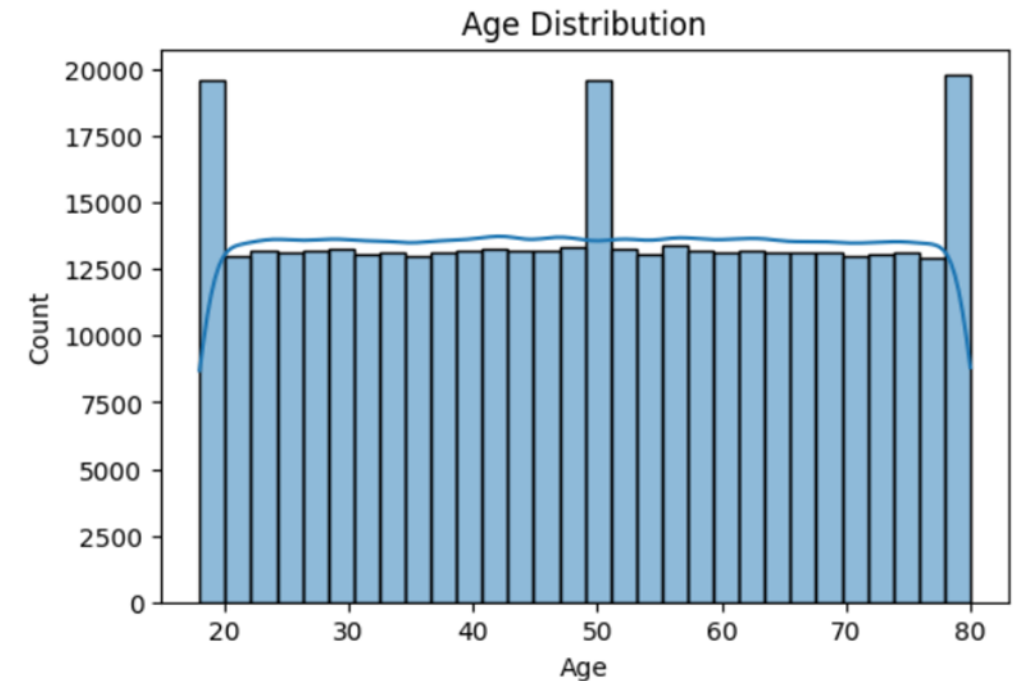
Best Decision Tree Parameters: {'criterion': 'entropy',
'max_depth': 20, 'min_samples_leaf': 2,
'min_samples_split': 5}

Accuracy: 63%

Methodology – Why Decision Tree With Pruning High Accuracy ??



- **Captures Non-Linear Relationships eg**
Age and Income :
- **Class Imbalance Handling via Splitting & SMOTE:**
- **Flexible Hyperparameters:**
- **No Overfitting** due to Flexible Hyperparameter
- **Improved Generalization and Robustness**
- **Handles Feature Interactions:** (via Entropy and Ginni)



Methodology – Naïve Bayes



Bernoulli Naive Bayes – 67%

- Optimized for binary features, it matches Gaussian Naive Bayes in performance.
- SMOTE helps in balancing the class distribution, further improving predictions.



Methodology – Random Forest



1. Random Forest (Default parameters) – 59%

- The default setup captures non-linear patterns but lacks parameter optimization, which limits its full potential.
- Handles feature interactions but suffers from overfitting with deep trees.

1. Random Forest (Tuned) – 61%

- Tuned parameters (max_depth, n_estimators) prevent overfitting and enhance generalization.
- Class weights ensure balanced predictions across minority and majority classes.

Why Random Forest (Tuned) > Random Forest (Default Parameter) ??

Methodology – Why the Tuned Model Performed Better (Accuracy: 61% vs. 59%)



Hyperparameter Tuning: The second model was tuned to avoid **overfitting**, ensuring it **Generalizes better on unseen data**.

Balanced Class Weights: **Improved handling of class imbalance** contributes to better performance, especially for minority classes.

More Trees: Increasing the number of trees to 200 stabilizes predictions by **Reducing variance**.

Depth Control: **Limiting the depth to 20** prevents the trees from becoming too complex, improving generalization.

MLP Classifier with Identity Activation – 58.7%

- Acts as a linear model, limiting the network's ability to learn non-linear patterns effectively.

MLP Classifier with Logistic/Tanh/ReLU Activation – 66-67%

- Non-linear activations (Logistic, Tanh, ReLU) enhance the network's capacity to learn complex relationships.
- ReLU prevents vanishing gradients, while Logistic and Tanh improve expressiveness, leading to higher performance.



Results (Midsem)



Machine Learning Technique	Accuracy(%)
Logistic Regression with Batch Gradient Descent (BGD)	57
Logistic Regression with Stochastic Gradient Descent (SGD)	52
L2 (Ridge) Logistic Regression with Recursive Feature Elimination (RFE)	57
Logistic Regression with PCA	54
Logistic Regression with SMOTE and OPTIMAL THRESHOLD	60
Random Forest (Default parameters)	59
Random Forest (Tuned)	61

Results (Midsem)

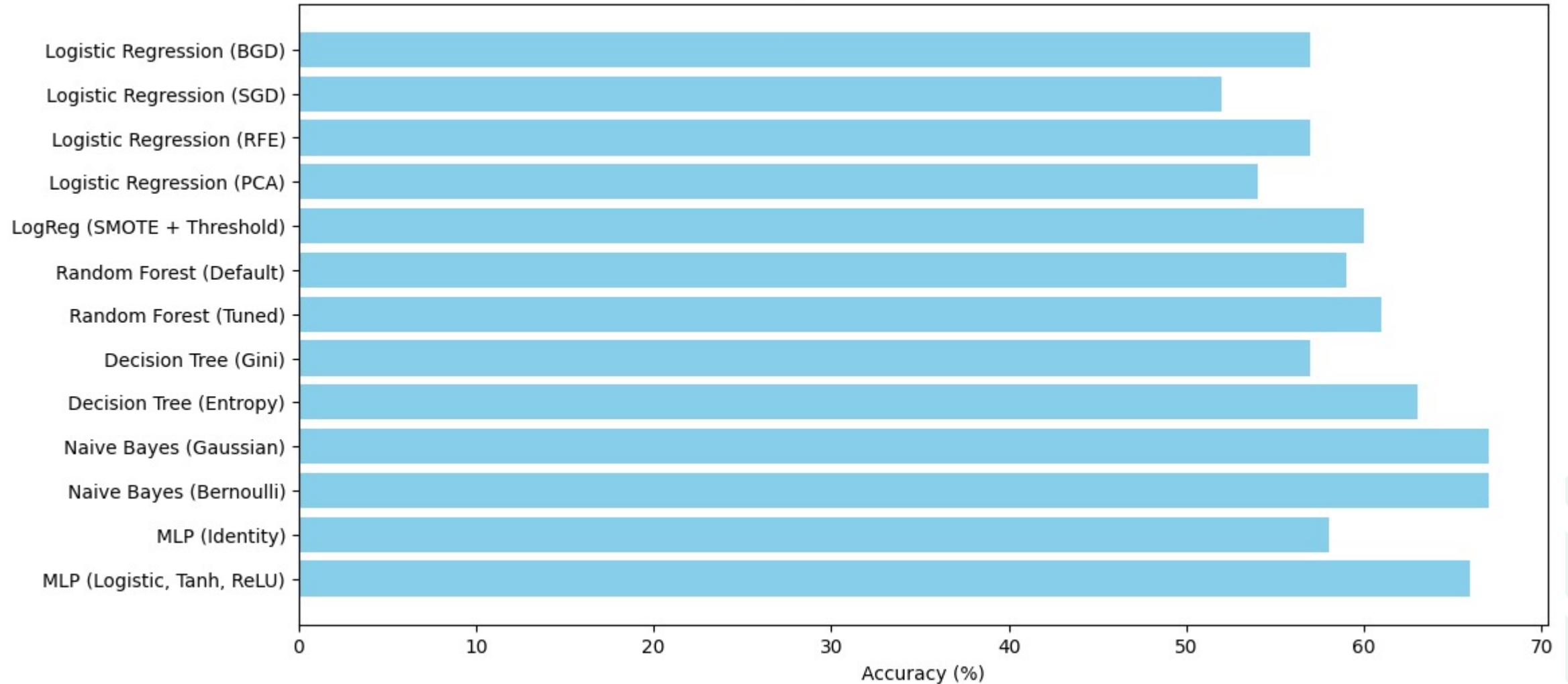


Decision Tree Classifier [CRITERIAN='GINI']	57
Decision Tree Classifier [CRITERIAN='ENTROPY']	63
Naive Bayes (Gaussian)	67
Naive Bayes (Bernoulli)	67
MLP Classifier('IDENTITY')	58
MLP Classifier('LOGISTIC','TANH','RELU')	66

Results (Midsem)



Comparison of Model Accuracies



Logistic Regression with SMOTE and Optimal Threshold vs. Basic Logistic Regression:

- Logistic Regression with SMOTE and threshold tuning (60%) outperforms the basic version (57%) by addressing class imbalance and refining decision boundaries, improving the separation between classes.

Tuned Random Forest vs. Default Random Forest:

- The tuned Random Forest (61%) performs better than the default version (59%) due to optimized hyperparameters (like `max_depth` and `n_estimators`), reducing overfitting and improving generalization..

Decision Tree (Entropy) vs. Decision Tree (Gini):

- The entropy-based Decision Tree (63%) achieves higher accuracy compared to Gini (57%) because it captures finer information gains during splits, enhancing class separation.

Analysis – Machine Learning Models



- **4) Naive Bayes vs. Logistic Regression:**
 - Naive Bayes (67%) performs better than Logistic Regression (57%) by efficiently handling the dataset's feature independence and class imbalance, making it well-suited for binary classification tasks.
- **5) MLP with Non-linear Activation vs. Identity Activation:**
 - MLP with non-linear activations (66-67%) outperforms identity activation (58%) as nonlinear functions like ReLU, Tanh, and Logistic allow the network to learn complex patterns, improving classification performance.



Report Link



Conclusion



- Naive Bayes (67%) performed best due to the dataset's alignment with the independence assumption and SMOTE balancing.
- Decision Tree (63%) captured non-linear patterns with well-tuned hyperparameters, while Random Forest (61%) handled feature interactions but struggled with variance. MLP (66-67%) utilized non-linear activations to learn complex patterns effectively.
- Logistic Regression with SMOTE and threshold tuning (60%) improved classification but fell short compared to non-linear models.

Overall, tuning, regularization, and class balancing were key to optimizing performance.



Future Works



- In future, we will apply SVM for better boundaries and ensemble methods like AdaBoost, XGBoost, and Voting Classifier to enhance accuracy and reduce bias. These techniques will help optimize predictions and handle complex patterns effectively.



Individual Member Contributions



All the members contributed equally to the work and helped each other in making edits to the codes and writing this report. The individual contributions listed below are only representations of the assignments of tasks to each team member.

- **Satyam:** Decision Tree, MLP
- **Aditya:** Logistic Regression ,Decision Tree
- **Suyash:** Random Forest, Model Evaluation
- **Vickey:** Data Preprocessing , Naive Bayes
- Satyam Pandey: Documentation and EDA



THANK YOU!

